IMPUTERS IMPROVE THE WRONG FEATURES: PLACEMENT, NOT RECONSTRUCTION ERROR ALONE, GOVERNS WHAT BETTER IMPUTATION BUYS

Anonymous authors
Paper under double-blind review

ABSTRACT

Imputers are compared by reconstruction error, and better reconstruction is assumed to buy better prediction. In practice the gain is small, and this paper measures where the improvement goes. Holding the total reconstruction error fixed and moving it across features changes downstream accuracy substantially, because predictive importance is concentrated in a minority of features, while the standard imputers we evaluate spread their improvement evenly and place only about a third of that improvement where the importance sits. We then attribute the gain using the imputations themselves: replacing only the important third's imputed values recovers most of what a better imputer is worth while spending a third of the error reduction, whereas the same total reduction placed on unimportant features recovers only a fraction. This holds at two missingness rates, on classification, and, with the sign reversed, under informative missingness, where better imputation instead costs tree ensembles. The diagnosis yields a procedure that imputes only where importance sits, at a fraction of the cost when the imputer is expensive and with no loss we can detect. Finally, treating imputer choice as a decision, we find that the diagnostic practitioners can compute is unreliable precisely when the missingness is informative, and is then beaten by a default fixed in advance. Better imputation is not failing to improve reconstruction; the improvement lands where the model cannot use it.

1 INTRODUCTION

The standard pipeline fills missing entries and then fits a model, and the filling step is chosen by how accurately it reconstructs held-out values. Le Morvan & Varoquaux (2025) showed that this buys less than expected: on a benchmark of numerical regression datasets with simulated missingness, large gains in imputation quality produce small gains in prediction, and in their Appendix L those gains shrink further and can reverse under MNAR, where they also cross imputation quality with the missingness indicator and attribute the reversal to mean imputation retaining the information that a value was imputed. Perez-Lebel et al. (2022) report the practical counterpart on real health databases: with gradient-boosted trees, mean imputation plus an indicator is hard to beat and native missing-value handling is better still.

What is missing from this picture is a mechanism for the *size* of the effect and a treatment of the decision a practitioner actually faces. Why do large reconstruction improvements produce small prediction improvements even under MCAR, where the missingness carries no information at all? And given that they do, how should one choose an imputer, and can the usual diagnostic support that choice? This paper answers both, and separates what we establish from what was already known.

(i) Where the improvement goes, and what it buys (§3). Holding the total reconstruction error fixed and moving it across features changes downstream accuracy substantially, because predictive importance is concentrated in a minority of features while the standard imputers we evaluate spread their improvement evenly, placing only about a third of it where that importance sits. We then close the loop with the imputations themselves rather than with injected noise: improving only the important third recovers most of the gain while spending a third of the reduction, whereas an equal total reduction placed on unimportant features recovers a fraction of it (Figure 3). This is a substantial

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

part of the explanation for why large reconstruction improvements yield small predictive gains, for the model families that gain at all.

(ii) Targeted imputation, the procedure the diagnosis implies (§4). Ranking columns by importance on a mean-imputed model and imputing only the top third with the expensive imputer matches imputing everything downstream in most (model, imputer) pairs, beats it in two and loses in one, at several times lower end-to-end cost when the imputer is expensive and at no saving when it is not. The diagnosis is actionable, with a stated condition.

(iii) Imputer selection, evaluated as a decision (§8). The diagnostic available on real data, reconstruction error on additionally masked *observed* entries, is accurate under MCAR and fails under MNAR, where it also reports its own error as far smaller than it is. The comparison that matters is not chance but a rule that needs no diagnostic: choosing the imputer that was best on *other* datasets, fixed before seeing the current one, is right far more often under MNAR and at a small fraction of the regret. Under MCAR that ordering reverses. On datasets that carry their own missing values, selecting by validation prediction instead of reconstruction does not change the conclusion, and the oracle ceiling over the same candidates, mean imputation included so that it is non-negative by construction, is small at the median with a long tail.

(iv) Replication and extension, credited as such (§6). The crossed imputation-by-indicator design under MNAR is due to Le Morvan & Varoquaux (2025), their Appendix L.2. We repeat it with paired per-dataset statistics, absolute scores for all four arms, two further tree families and a linear model, and quantify the recovery: the MNAR penalty those tree families pay is removed by supplying the mask, on most datasets, and the gain that a linear model retains under MNAR is itself largely substitutable by the mask. We claim the quantification and the breadth, not the design.

We also report model-family differences under uninformative missingness (§5) with explicit controls, and we state where our evidence does not reach: in particular an association between a dataset's missingness informativeness and the loss that better imputation causes trees is visible but not statistically established (§7).

2 SETUP

Data. Simulated-missingness experiments use the 19 numerical regression datasets of Le Morvan & Varoquaux (2025) (OpenML (Vanschoren et al., 2014) suite 336) and the 15 numerical binary-classification datasets of suite 337 (Grinsztajn et al., 2022) (dropping *Bioresponse*, whose feature count $d = 419$ makes it costly); rows capped at 5000, splits $70/15/15$. Natural-missingness experiments use 23 datasets that arrive with missing values: five clinical (ADNI (Petersen et al., 2010) with a diagnosis and an ADAS-Cog target, ABIDE (Di Martino et al., 2014), a cervical-cancer screening cohort (Fernandes et al., 2017), a cirrhosis cohort derived from the Mayo Clinic trial data of Fleming & Harrington (2013)) and 18 public OpenML datasets including SUPPORT2 (Knaus et al., 1995); identifiers, targets and exclusions are in Appendix A. Five of the public datasets were added in a later screening pass that applied an automatic guard (a dataset is discarded if mean imputation alone already reaches AUC > 0.99 or $R^2 > 0.99$, the failure mode we had to fix by hand for ADNI and ABIDE), and they include the widest cases in the collection ($d = 113$ to 139).

Missingness, models, imputers. Simulated masks follow the standard taxonomy (Rubin, 1976): MCAR, MAR (logistic) and MNAR (self-masking), at 20% and 50%, drawn on training-standardised features, with the benchmark's preprocessing reproduced. Models: MLP, FT-Transformer (Gorishniy et al., 2021), SAINT (Somepalli et al., 2021), XGBoost (Chen & Guestrin, 2016), random forest (Breiman, 2001), ridge/logistic regression and HistGradientBoosting; each tunable family gets 50 Optuna (Akiba et al., 2019) trials per cell on the cell's complete data, so no imputer is favoured, with untuned defaults reported alongside. Imputers: mean, KNN (Troyanskaya et al., 2001), iterative (Van Buuren & Groothuis-Oudshoorn, 2011) and random-forest imputation (Stekhoven & Bühlmann, 2012), together with the Gaussian conditional expectation where the generating process allows it. The selection study of §8 enlarges the candidate pool to nine, adding median imputation, two further neighbourhood sizes, iterative imputation with extremely randomised trees, and low-rank completion.

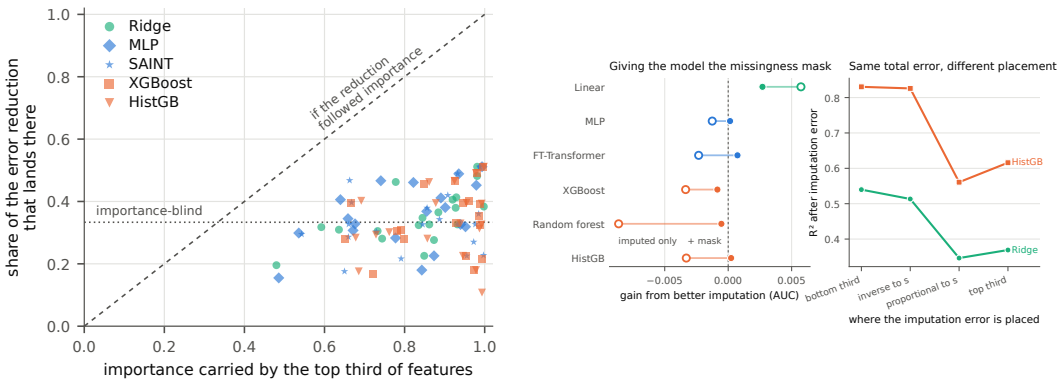


Figure 1: **Left:** for each dataset and model, the share of predictive importance carried by the top third of features (horizontal) against the share of a real imputer's error reduction that lands there (vertical); the dashed diagonal marks proportional placement, the dotted line an importance-blind one. **Right:** the same total reconstruction error moved across features by hand (§3); and, where the missingness is real, the effect of supplying the mask (§7).

Estimand and statistics. Unless stated otherwise, gain $=$ score$(I^\star)-$score(I_{mean}) with $I^\star$ selected by reconstruction error: on simulated data on the training rows' actually missing entries, on real data on an extra 10% of observed training entries; §8 varies the rule. Five seeds, averaged first; all tests at the dataset level ($n = 19$, 15 or 23), two-sided Wilcoxon signed-rank (Wilcoxon, 1992) with Holm correction (Holm, 1979) within each table, and 2000-resample bootstraps over datasets (Efron, 1992). Exact two-sided floors: 3.8×10^{-6} ($n = 19$), 6.1×10^{-5} ($n = 15$), 2.0×10^{-3} ($n = 10$), 7.8×10^{-3} ($n = 8$).

3 Where the Error Is, and Where the Improvement Goes

The metric does not determine the consequence. We inject Gaussian error at the missing entries with a fixed pooled budget and vary only its allocation (uniform, proportional to permutation importance, inverse to it, or concentrated on the most or least important third), then refit. At identical pooled reconstruction error the downstream R^2 spans a median of 0.12 (ridge) to 0.20 (MLP, SAINT), up to 0.72, exceeding 0.10 in 63–83% of (dataset, model, budget) cells. Importance is concentrated: the bottom third of features carries a median 0.80% of it and the top third 84–94%. These figures use permutation importance of a model fitted to the complete data; on the mean-imputed model that a practitioner would have, the top third carries 72–84%, a flattening consistent with a feature mattering less once its values are imputed.

The imputers we evaluate spend their improvement without regard to importance. The intervention above is artificial by construction, so it cannot by itself explain the imputers practitioners use. We therefore decompose the reduction achieved by the candidates we evaluate. For every cell we compute each imputer's per-feature mean squared error on the entries that are missing, take the best-reconstructing imputer, and ask where its improvement over mean imputation falls. The answer is: everywhere equally. A median 33–34% of the total reduction lands on the top third of features, which carry 84–95% of the importance; the importance-weighted reduction exceeds an importance-blind allocation by 0–13% at the median across the five model families, the largest being ridge's 1.13 (Figure 1, left; per-family medians in Appendix E, Table 9). These are medians within each family; individual datasets range from 0.11 to 0.51 (Figure 1).

What that improvement is worth: a direct attribution. The intervention above uses injected Gaussian error, so it cannot settle what the *real* improvement is worth. We therefore use the imputations themselves. Starting from the mean-imputed matrix we replace the imputed values of a chosen subset of features with the better imputer's values and refit: the important third alone; the remaining two thirds alone; and, as the control that matters, a set of *least* important features chosen

Table 1: Attribution of a real imputer's gain (medians over 19 datasets, 5 seeds, MCAR 50%; importance from the mean-imputed model, §3). "Δerr share" is the fraction of the imputer's total per-feature error reduction the arm uses. Paired dataset-level Wilcoxon against the important-third arm, Holm-corrected across the four families: against the same-Δerr control, $p < 10^{-4}$ (ridge), 0.0013 (MLP), 0.52 (XGBoost), 0.77 (HistGB); against every feature, $p < 10^{-4}$, 0.0016, 1.00, 1.00.

	every feature		important third		same Δerr, unimportant	
model	Δerr share	gain	Δerr share	gain	Δerr share	gain
Ridge	1.00	+0.0505	0.32	+0.0437	0.36	+0.0068
MLP	1.00	+0.0328	0.31	+0.0229	0.36	+0.0100
XGBoost	1.00	+0.0116	0.35	+0.0110	0.37	+0.0045
HistGB	1.00	+0.0033	0.35	+0.0028	0.39	+0.0045

greedily so that its total error reduction matches that of the important third. Everything else, including the model, the split and the total amount of missingness, is held fixed.

The control is if anything favoured: because features are added greedily until the target is reached, its total reduction is never smaller than the important third's, exceeding it by a median factor of 1.12–1.15 (10th–90th percentile 1.03–1.85). For the two families that gain from better imputation at all, a third of the reduction placed on the important third delivers most of the gain (87% for ridge and 70% for the MLP), while the same total reduction placed on unimportant features delivers 13% and 30%. The paired contrast between those two arms is +0.0153 R^2 for ridge (95% CI $[+0.0056, +0.0369]$) and +0.0065 for the MLP ($[+0.0034, +0.0148]$), Holm $p \leq 0.0013$; restricting to the datasets where the two arms' reductions match within 15% gives +0.0349 ($n = 9$, $p = 0.004$) and +0.0046 ($n = 11$, $p = 0.042$) (Table 1, Figure 3). We are explicit about the limits of this: the important third does not account for *all* of the gain (the remaining two thirds are worth +0.017 and +0.011, and the difference between "important third" and "every feature" is significant), and for XGBoost and HistGB the total gain is so small (+0.012 and +0.003) that none of the contrasts resolve. The claim we support is therefore narrow and explicit: in the MCAR-50% regression setting, for the two families whose gains are large enough to decompose, most of what the improvement is worth comes from where it lands. It is not a general account of the absolute size of gains, nor of the differences between families, which §9 reports as unresolved.

How far the attribution extends. The experiment above is run at MCAR 50% on the regression suite, where two families have gains large enough to decompose. Repeating it in three further regimes (Appendix E, Table 6) shows the same structure and widens it. At 20% missingness, where every family gains more because more of each row is observed to condition on, the important third recovers 68–103% of the full gain and the contrast against the matched-reduction control is significant for *all three* families tested, gradient-boosted trees included (+0.0125 HistGB, +0.0419 ridge, +0.0077 XGBoost; Holm $p \leq 0.0048$). On the classification suite the linear model behaves the same way (+0.0067 AUC, Holm $p = 0.050$) while the tree families have no positive gain to decompose. Under MNAR the signs invert but the logic holds: better imputation *hurts* trees, and the harm too is concentrated on the important third (−0.0219 XGBoost, Holm $p = 0.011$; −0.0201 HistGB, $p = 0.025$, against the matched control). Replacing the constant fill where the model looks is what destroys the most information. Placement governs what better imputation is worth in every regime we examined; what changes with the regime is the sign. Off MCAR the control is matched on the unweighted reduction, which need not equal the weighted pooled one (Appendix B).

Would weighting the metric by importance help? Since placement governs value, an obvious repair is to score imputers by importance-weighted reconstruction error. We tested that directly rather than by proxy: for every cell we rank the four candidates by pooled error and by importance-weighted error, and compare the downstream score of the imputer each rule selects. The two rules select different imputers in 16.2% of cells, so weighting does re-rank, contrary to what the collinearity of the error profiles (median cosine similarity 0.97 across candidates, against 0.39 for deliberately varied placements) might suggest. Yet the rankings are otherwise identical (median Kendall $\tau = 1.00$), and the change does not pay: the downstream difference is +0.0000 R^2 at the median overall and +0.0006 ($p = 0.125$) even when restricted to the (dataset, model) pairs where the choice actually

Table 2: Targeted imputation against full imputation (medians over 19 datasets, 5 seeds, MCAR 50%). "step" is the speed-up of the imputation step alone; "net" includes the importance ranking the targeted pipeline requires. The difference column is the median of the per-dataset paired differences and need not equal the difference of the two medians shown.

imputer	model	gain full	gain targeted	diff. (targ.−full)	p	step	net
random forest (6.2 s)	Ridge	+0.0511	+0.0645	−0.0061	0.196	13.4×	12.3×
	MLP	+0.0268	+0.0212	−0.0048	0.045	13.9×	4.0×
	XGBoost	+0.0057	+0.0059	−0.0020	0.953	13.5×	5.3×
	HistGB	+0.0031	+0.0023	−0.0010	0.768	13.3×	5.8×
iterative (0.18 s)	Ridge	+0.0248	+0.0329	−0.0001	0.709	31.4×	5.5×
	MLP	+0.0182	+0.0117	−0.0064	0.418	29.2×	0.2×
	XGBoost	−0.0046	+0.0025	+0.0094	0.049	26.3×	0.3×
	HistGB	−0.0115	+0.0004	+0.0112	0.049	29.7×	0.4×

changes. Across datasets, the importance-weighted reduction also predicts the gain no better than the pooled one in a way we can resolve (differences in Spearman ρ of -0.00 to $+0.14$, all bootstrap intervals including zero, Appendix E). The scoped conclusion is that among these candidates the weighted metric changes the choice occasionally and the outcome negligibly; the reason is not that placement does not matter (Table 1 shows it does) but that the candidates barely differ in it.

Does this depend on how importance is defined? Permutation importance behaves awkwardly under correlated features, so we repeated both arms under drop-column, conditional-permutation and model-native rankings and a random control (Table 11). For ridge, where the contrast resolves, the four definitions give $+0.0153$, $+0.0151$, $+0.0153$ and $+0.0141$ R^2 ($p \leq 10^{-4}$); a random ranking gives -0.0036 ($p = 0.11$). The claim is about predictive importance, not about one estimator of it.

How these quantities are computed. Per-feature error is an imputer's mean squared error on the entries that are actually missing, on training rows; the reduction is $\Delta_j = e_j^{\text{mean}} - e_j^{\text{best}}$, signed; importance is permutation importance of the model a practitioner would have, fitted on mean-imputed data and permuted on the validation split. Appendix B gives the full definitions, the weighting, and the robustness checks (clipping negative reductions changes the shares by at most 0.007; using complete-data importance instead leaves the attribution intact).

4 Targeted Imputation: Spend the Imputer Where It Pays

The diagnosis suggests a procedure, and the procedure is cheap. Rank features by permutation importance on a model fitted to the mean-imputed training data (one model fit, no complete data required), then run the expensive imputer *only* for the top third of columns, predicting each from the mean-filled remainder in a single pass, leaving the other columns mean-imputed. Unlike the attribution arms of §3, this never runs the full imputer: its cost scales with the number of columns one chooses to model, not with d.

Table 2 and Figure 2 give the result on the 19 regression datasets. Downstream, no difference from imputing everything is detected in five of the eight (model, imputer) pairs; it is *better* in two and worse in one. The two are the tree families with the iterative imputer, where imputing every column costs them (-0.0115 and -0.0046 R^2) and the targeted version does not ($+0.0004$ and $+0.0025$, paired $p = 0.049$); the one is the MLP with the random-forest imputer (-0.0048, $p = 0.045$).

When it actually saves time. The imputation step itself becomes 13× cheaper with the random-forest imputer and 26–31× cheaper with the iterative one, but the procedure also needs an importance ranking, and that costs one model fit and one permutation pass: a median 0.02 s for ridge, 0.40 s for HistGB, 0.77 s for the MLP and 0.94 s for XGBoost. Counting it, the picture splits cleanly by how expensive the imputer is. Against the random-forest imputer (median 6.2 s per cell) the net saving is 4.0–12.3× and the targeted pipeline is faster on 89–100% of datasets; the advantage grows with the number of columns, from 3.4× at $d \leq 10$ to 11.2× above $d = 50$. Against the iterative BayesianRidge imputer (median 0.18 s) the ranking dominates, and targeting is a net loss for

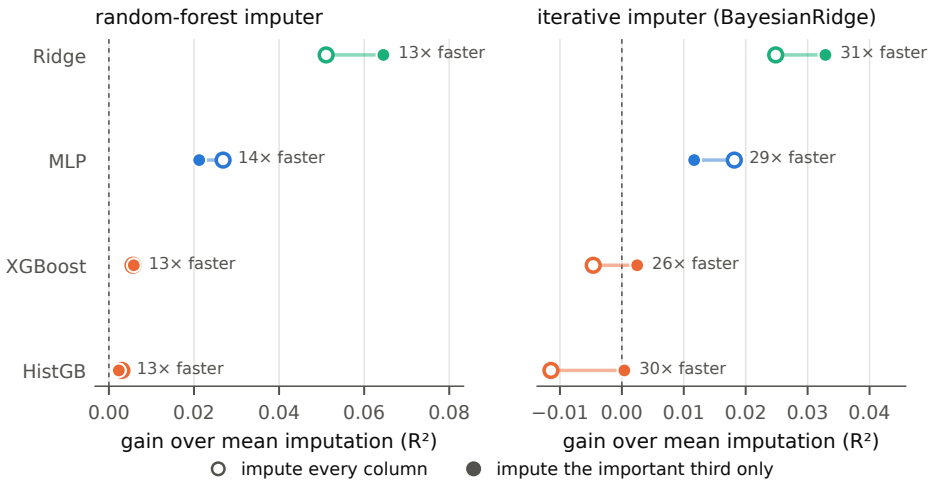


Figure 2: Targeted imputation. Open circles: the expensive imputer run over every column. Filled: the same imputer run only for the most important third of columns, with mean imputation elsewhere. Medians over 19 datasets, 5 seeds; annotations give the median speed-up of the imputation step.

every downstream model except ridge, whose ranking is nearly free. The prescription is therefore conditional and we state it that way: targeting is worth it when the imputer is expensive relative to a single model fit (the MissForest-style case, which is also the case where practitioners hesitate), and pointless when imputing everything already costs less than ranking the columns.

Beyond regression under MCAR. Repeating the procedure on the classification suite and on the natural-missingness datasets (Appendix E, Table 8) makes the case stronger rather than weaker: there, targeting is significantly *better* than imputing every column for the tree families, by $+0.0066$ AUC for XGBoost and $+0.0052$ for HistGB with the random-forest imputer ($p \leq 0.018$), $+0.0069$ for the random forest with the iterative one ($p = 0.008$), and $+0.0016$ on the natural-missingness datasets ($p = 0.003$); full imputation hurts those families and targeting limits the damage. Net speed-ups there are 0.1–$11.3\times$, the low end being the cheap iterative imputer, for which targeting is a net loss as in the regression setting. Repeating both settings at $k/d \in \{1/6, 1/2, 2/3\}$ leaves the paired difference positive for every tree family at every budget with the random-forest imputer, so the conclusion does not rest on the $k = d/3$ used throughout.

Two further caveats. The median paired differences straddle zero in the regression/MCAR setting, so that part is a cost result with no detected loss, not a demonstration of equivalence: a two-sided interval excludes a 0.01 R^2 difference in only two of the eight pairs (Appendix E). And where targeting *helps* a tree ensemble, two things change at once: fewer columns, and a single pass instead of ten. A third arm that keeps the full schedule but updates only the selected columns splits the advantage about evenly (HistGB $+0.0112$ total, $+0.0056$ columns, $+0.0053$ schedule; XGBoost $+0.0094$, $+0.0002$, $+0.0037$), neither component significant at $n = 19$. The ranking is separable: against a random subset of the same size that arm is worth $+0.0120$ R^2 for ridge ($p = 0.0024$) and nothing for the trees (Appendix E).

5 Model Families Under Uninformative Missingness

The gain from better imputation also differs by model family: under MCAR all nine tuned neural-vs-tree pairs favour the neural side in regression ($+0.014$ to $+0.030$ R^2, 79–95% of datasets, Holm $p \in [0.0002, 0.019]$) and all six do in classification (87–100% of datasets, Holm $p \leq 0.0043$), and untuned defaults show the same gap (Appendix C). The expressiveness account is not settled by our controls: adding a measure of each configuration's ability to absorb imputation error to the usual control for clean accuracy halves the difference and, in regression, leaves it indistinguishable from zero ($+0.0118$, $[-0.0007, +0.0242]$; classification $+0.0100$, $[+0.0058, +0.0150]$). Within families,

Table 3: Selection rules by mechanism, over the four-candidate pool and the nine-candidate pool: top-1 accuracy of the held-out-entry diagnostic, its mean relative regret in true reconstruction error, and the regret of the leave-one-dataset-out default that uses no diagnostic.

	four candidates			nine candidates		
	diag. top-1	diag. regret	default regret	diag. top-1	diag. regret	default regret
MCAR	87.4%	0.012	0.025	81.1%	0.013	0.019
MAR	70.5%	0.063	0.066	65.3%	0.073	0.051
MNAR	49.5%	0.112	0.006	34.7%	0.108	0.066

gains fall as capacity rises, the direction that account predicts. We report the ordering without claiming to explain it; §9 lists the explanations we rejected.

6 Informative Missingness: Replication with Paired Statistics

Under MNAR the sign of the effect changes, which matters for §4 because it sets when imputation should not be bought at all. The crossed design that establishes this (imputation quality against the presence of the missingness mask) is due to Le Morvan & Varoquaux (2025) (their Figures 30 and 34); we repeat it on our grid with the imputer selected by its error on the actually missing entries, and add paired per-dataset statistics, absolute scores and three further model families. Without the mask, better imputation costs HistGB -0.034, random forest -0.034 and XGBoost -0.046 R^2; with it, -0.005, -0.002 and $+0.004$, so $+0.028$ to $+0.039$ is recovered on 84–100% of datasets (Holm $p \leq 0.0004$). The absolute scores show restoration rather than compression: for HistGB the better imputer moves from 0.542, below mean imputation's 0.563, to 0.564 beside mean-plus-mask at 0.569. Under MCAR the same intervention recovers an order of magnitude less ($+0.004$ to $+0.006$), and the mask is worth $+0.0008$ there against $+0.005$ under MNAR: the effect tracks how informative the mask is, not the mechanism's name. Ridge keeps its positive MNAR gain without the mask ($+0.084$) but only $+0.017$ of it once the mask is present, which alone is worth $+0.122$ R^2 to it: what better imputation buys a linear model under MNAR is largely what the mask supplies directly.

7 Datasets Whose Missingness Is Real

On 23 datasets that carry their own missing values (12 regression, 11 classification; Appendix A), typical gains from the reconstruction-selected imputer are small and mostly negative: in regression every configuration's median lies within ± 0.009 R^2, in classification the linear model gains $+0.0013$ AUC (positive on 73% of datasets, $p = 0.014$) while tree ensembles lose (-0.002 to -0.004; random forest $p = 0.005$). Two controls hold: per-imputation tuning moves gains by at most ± 0.002, and degradation under injected error still separates the families (MLP vs. XGBoost, $+0.012$ R^2, Holm $p = 0.023$), so the models are sensitive to imputation error, just not to what separates them.

With the enlarged collection one contrast does resolve, and it runs against the MCAR result of §5: at matched base accuracy, neural networks gain *less* than tree ensembles here, -0.0230 R^2 (95% CI -0.0397 to -0.0029, $n = 12$) in regression, with the classification estimate still indistinguishable from zero (-0.0025, $[-0.0104, +0.0031]$, $n = 11$). The reversal is carried by the FT-Transformer (-0.0085 median, positive on one of twelve datasets, $p = 0.009$), and is consistent with §6: the models that exploit the constant fill most have most to lose from replacing it.

An association we can see but not establish. Given §6, one expects better imputation to cost tree ensembles more where the missingness pattern is more predictive. We measure that pattern's content independently, by fitting a model on the mask alone and scoring it against a no-feature baseline. The association has the predicted sign but is not significant (Spearman -0.24, $n = 10$ regression; -0.36, $n = 8$ classification; Holm $p = 1$ across the six family-by-task correlations; leave-one-dataset-out ranges $[-0.55, +0.05]$ and $[-0.61, -0.11]$). A stronger-looking -0.76 ($p = 0.028$) is obtained if informativeness is measured by the accuracy the mask adds, but that conflates the pattern's content with what the baseline exploits, so we report the association as exploratory.

8 CHOOSING AN IMPUTER

The diagnostic, and a rule that does not need one. On real data, reconstruction quality can only be estimated by masking additional observed entries, which is valid when observed and missing entries are exchangeable and false under MNAR. We evaluate selection rules by top-1 accuracy *and* by regret in true reconstruction error, on simulated cells where both are computable. The comparison rule is fixed before the dataset it is applied to: use whichever imputer reconstructs best on the *other* 18 datasets, by majority over their per-cell winners (Table 3).

The pattern reproduces when the candidate pool is more than doubled, from mean, KNN, iterative and random-forest imputation to nine methods spanning simple, neighbour-based, iterative (BayesianRidge, random forest, extremely randomised trees) and low-rank completion. Under MCAR the diagnostic is worth using: its regret is below the fixed default's with either pool. Under MNAR it is not: its regret is two to eighteen times the default's, and its top-1 accuracy falls further as the pool grows, since the diagnostic is miscalibrated there, reporting errors at a third of their true size. Two honest qualifications follow. The *identity* of the transferable default is a property of the pool, not a discovery about imputation: with four candidates it is the random-forest imputer, with nine it is the extremely-randomised-trees variant, which is the true best in 78% of MCAR cells. And top-1 accuracy is the wrong summary when candidates are close: under MNAR with nine candidates the default's top-1 is only 13.7% because two imputers are near-tied (48% each), yet its regret is the lower of the two rules. Regret, not accuracy, is what should be read from this table.

Downstream the cost of those mistakes is small (median regret 0.000 R^2): the signal is least reliable where the choice matters least. On the 23 natural-missingness datasets, where the mechanism is unknown, the same leave-one-dataset-out construction selects *mean imputation* for nine of twelve (task, model) combinations and beats reconstruction-based selection on ten of twelve, significantly for the FT-Transformer in regression ($+0.0085$ R^2, $p = 0.009$) and the random forest in classification ($+0.0042$ AUC, $p = 0.005$). The rule that transfers is population-dependent; that a fixed rule beats the diagnostic outside MCAR is what reproduces.

Selecting by prediction instead. With labels, one can fit each candidate pipeline and select on validation score. This changes the rule, not the conclusion (Table 10). Across six families on the natural-missingness datasets, the only configuration with a positive validation-selected gain in both tasks is the linear model ($+0.008$ AUC on 100% of the eight classification datasets; raw $p = 0.008$, the exact floor at $n = 8$, and $p = 0.14$ after Holm correction across that table). Tree ensembles lose (-0.006 to -0.009 AUC); allowing mean imputation itself into the comparison returns every tree ensemble to 0.000. Oracle selection on the test set (over the same four candidates, mean imputation included, so that the ceiling is non-negative by construction) is a median $+0.003$ to $+0.011$ depending on the family. That median hides a tail: the ceiling exceeds $+0.01$ on 0–55% of datasets depending on the family and task, its upper quartile reaches $+0.019$, and on one regression dataset it is $+1.13$ R^2. That dataset is the one where *mean* imputation itself fails: on `all-natural-disasters`, XGBoost scores -2.41 R^2 with mean imputation and $+0.79$ with the random-forest imputer, so the oracle buys escape from a catastrophic baseline rather than a refinement among good options. Imputer choice usually buys little and occasionally a great deal, which argues for checking it cheaply on a validation split rather than for ignoring it. These ceilings bound only the candidates we evaluate.

What a practitioner should do with a tree ensemble. Targeting assumes one has to impute; gradient-boosted trees do not. Comparing mean imputation, the selected imputer, targeted imputation, mean-plus-indicator and native handling on the same 23 datasets (Table 12), native handling wins: it beats targeted imputation by $+0.0041$ and $+0.0030$ AUC ($p = 0.024$, $p = 0.032$) and by $+0.0053$ R^2 for XGBoost ($p = 0.052$), at no imputation cost. The prescription of §4 applies where imputation is unavoidable (linear models, MLPs, transformers), not to trees that consume missingness directly.

Native handling also matches mean-plus-indicator and is unchanged by appending the mask, because the routing already encodes it; this reproduces Perez-Lebel et al. (2022) within our grid. Imputation cost about 68 CPU-seconds per cell here, against nothing for native handling.

9 What Does Not Work

Several natural repairs fail, and we report them so they are not retried: restoring the imputed distribution by stochastic draws or variance correction (worse in 75–100% of cells, $p < 10^{-5}$), a sensitivity×imputability product, mask-interaction features, and target-aware (pseudo-label) imputation, which degrades imputation itself. Under MCAR we also could not establish *why* tree ensembles benefit least: saturation, headroom compression, feature substitution through correlated splits, a worse sensitivity-ranking signal and sub-additive damage were each tested and rejected. Effect sizes and the tests behind both lists are in Appendix E.

10 Related Work

Le Morvan & Varoquaux (2025) is our closest reference: we adopt its datasets, architectures, preprocessing and protocol, its Appendix L.2 contains the crossed imputation-by-indicator design we repeat in §6, and its MNAR observation is the finding we quantify per dataset rather than discover. Perez-Lebel et al. (2022) established on real health records that simple strategies are hard to beat for gradient-boosted trees, which §8 reproduces on a broader, partly non-clinical collection. Theory explains why the mask matters: the Bayes-optimal predictor depends on the observed values and the missingness pattern, and architectures consuming the pattern are robust to mechanisms that defeat impute-then-predict (Le Morvan et al., 2020; 2021), while impute-then-predict can be consistent given a powerful enough learner (Josse et al., 2024). MIA (Twala et al., 2008) and the missing-indicator method (Little & Rubin, 2019) are the classical alternatives; Stempfle et al. (2025) make avoidance a learning objective and Bertsimas et al. (2025) adapt the predictor to whichever features were observed, both learning-side counterparts to §8. What importance means under missingness is itself contested (Von Kleist et al., 2025): ours is importance under the *observed* data, on the mean-imputed matrix a practitioner holds, with four alternatives in Tables 9 and 11. Shadbahr et al. (2023) likewise find entry-wise error the wrong score for downstream value and propose distributional discrepancy; we locate the gap in where across features the error sits relative to importance. The observation that imputers treat features as equally worth recovering while downstream importance is unequal has also motivated methods: Guo et al. (2024) weight a matrix-completion objective by feature importance, from the same premise as ours. They propose an objective and show it predicts better; we propose none, and instead measure how much of a standard imputer's *realised* improvement reaches important features and how much of the gain that accounts for against a reduction-matched control. Deciding *which* variables to impute at all has been raised outside prediction too: Berrevoets et al. (2023) argue for selective imputation in treatment-effect estimation. Those lines change the imputer's objective or the estimand; ours is measurement-side and then procedural, adding that reweighting the *selection* metric over existing candidates does not help and that the diagnosis becomes a targeting rule with a stated cost condition.

11 Conclusion

Better imputation is not failing to improve reconstruction; the imputers we evaluate improve it in the wrong places, spreading gains evenly over features whose importance is not. That is a major reason gains are small even under MCAR, and when the missingness is informative the same placement logic turns the gain into a loss, because the constant fill it replaces was itself carrying the pattern. The choice between imputers inherits the blindness: the diagnostic a practitioner can compute is least reliable exactly where the mechanism is worst, and a default fixed in advance on other datasets then picks better, though what that buys downstream is usually small. When a complete matrix is required, imputing only where importance sits costs a fraction as much for the same result; when the model can route missing values itself, the right move is to let it. Two boundaries hold throughout: the natural-missingness tests have limited power ($n = 12$ and $n = 11$ tasks), so results there are measurements rather than demonstrations of absence, and the selection ceilings are properties of the candidate pool we evaluate rather than of imputation in general.

AI USE STATEMENT

An LLM-based assistant was used for coding, running experiments, formatting LaTeX tables, and editing the manuscript. The authors independently verified all experimental results and take full responsibility for the final content.

ETHICS STATEMENT

The natural-missingness collection draws on four clinical cohorts. ADNI and ABIDE are de-identified at source and were obtained from their official distributions under the data use agreements their custodians require; the cervical-cancer screening and cirrhosis cohorts are public research releases. We redistribute no participant records, only loaders that rebuild the feature tables from data obtained the same way, and we make no clinical claim.

REPRODUCIBILITY STATEMENT

Datasets, targets and exclusions are in Appendix A, the protocol (tuning, the placement intervention, the importance and reduction definitions, and imputer selection) in Appendix B, the model-family comparison in Appendix C, absolute scores for the crossed design in Appendix D, the further regimes and budgets in Appendix E, and two fixes to the published benchmark's code that our experiments required in Appendix F. Every table and figure is computed by one script from released per-cell result files, so all of them can be reproduced without retraining. Anonymized code to reproduce our experiments, together with those result files and the tuned hyperparameters, is provided in the supplementary material.

REFERENCES

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. pp. 2623–2631, 2019.

Jeroen Berrevoets, Fergus Imrie, Trent Kyono, James Jordon, and Mihaela Van der Schaar. To impute or not to impute? missing data in treatment effect estimation. In *International Conference on Artificial Intelligence and Statistics*, pp. 3568–3590. PMLR, 2023.

Dimitris Bertsimas, Arthur Delarue, and Jean Pauphilet. Adaptive optimization for prediction with missing data. *Machine Learning*, 114(5):124, 2025.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6):659–667, 2014.

Bradley Efron. Bootstrap methods: another look at the jackknife. pp. 569–593, 1992.

Kelwin Fernandes, Jaime S Cardoso, and Jessica Fernandes. Transfer learning with partial observability applied to cervical cancer screening. In *Iberian conference on pattern recognition and image analysis*, pp. 243–250. Springer, 2017.

Thomas R Fleming and David P Harrington. *Counting processes and survival analysis*. John Wiley & Sons, 2013.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. volume 34, pp. 18932–18943, 2021.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? volume 35, pp. 507–520, 2022.

Cong Guo, Wei Yang, Chun Liu, and Zheng Li. Iterative missing value imputation based on feature importance: C. guo et al. *Knowledge and Information Systems*, 66(10):6387–6414, 2024.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pp. 65–70, 1979.

Julie Josse, Jacob M Chen, Nicolas Prost, Gaël Varoquaux, and Erwan Scornet. On the consistency of supervised learning with missing values. *Statistical Papers*, 65(9):5447–5479, 2024.

William A Knaus, Frank E Harrell, Joanne Lynn, Lee Goldman, Russell S Phillips, Alfred F Connors, Neal V Dawson, William J Fulkerson, Robert M Califf, Norman Desbiens, et al. The support prognostic model: Objective estimates of survival for seriously ill hospitalized adults. *Annals of internal medicine*, 122(3):191–203, 1995.

Marine Le Morvan and Gaël Varoquaux. Imputation for prediction: beware of diminishing returns. In *International Conference on Learning Representations*, volume 2025, pp. 41174–41213, 2025.

Marine Le Morvan, Julie Josse, Thomas Moreau, Erwan Scornet, and Gaël Varoquaux. Neumiss networks: differentiable programming for supervised learning with missing values. volume 33, pp. 5980–5990, 2020.

Marine Le Morvan, Julie Josse, Erwan Scornet, and Gaël Varoquaux. What'sa good imputation to predict with missing values? volume 34, pp. 11530–11540, 2021.

Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2019.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Alexandre Perez-Lebel, Gaël Varoquaux, Marine Le Morvan, Julie Josse, and Jean-Baptiste Poline. Benchmarking missing-values approaches for predictive models on health databases. *GigaScience*, 11:giac013, 2022.

Ronald Carl Petersen, Paul S Aisen, Laurel A Beckett, Michael C Donohue, Anthony Collins Gamst, Danielle J Harvey, Clifford R Jack Jr, William J Jagust, Leslie M Shaw, Arthur W Toga, et al. Alzheimer's disease neuroimaging initiative (adni) clinical characterization. *Neurology*, 74(3): 201–209, 2010.

Donald B Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

Tolou Shadbahr, Michael Roberts, Jan Stanczuk, Julian Gilbey, Philip Teare, Sören Dittmer, Matthew Thorpe, Ramon Viñas Torné, Evis Sala, Pietro Lió, et al. The impact of imputation quality on machine learning classifiers for datasets with missing values. *Communications medicine*, 3(1):139, 2023.

Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C Bayan Bruss, and Tom Goldstein. Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv preprint arXiv:2106.01342*, 2021.

Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

Lena Stempfle, Anton Matsson, Newton Mwai, and Fredrik D. Johansson. Prediction models that learn to avoid missing values. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 56899–56919. PMLR, 13–19 Jul 2025.

Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

Bheki ETH Twala, MC Jones, and David J Hand. Good methods for coping with missing data in decision trees. *Pattern Recognition Letters*, 29(7):950–956, 2008.

Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011.

Joaquin Vanschoren, Jan N Van Rijn, Bernd Bischl, and Luis Torgo. Openml: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, 15(2):49–60, 2014.

Henrik Von Kleist, Joshua Wendland, Ilya Shpitser, and Carsten Marr. Feature importance metrics in the presence of missing data. In *ICML*, 2025.

Frank Wilcoxon. Individual comparisons by ranking methods. pp. 196–202, 1992.

A DATASETS

Simulated missingness. OpenML suite 336 (19 numerical regression datasets) as used by Le Morvan & Varoquaux (2025), and suite 337 (Grinsztajn et al., 2022) excluding *Bioresponse*.

Natural missingness. Clinical: ADNI3 (baseline biomarkers → `DX_bl`, and → `ADAS13_bl`), ABIDE (`DX_GROUP`), cervical-cancer screening (`Biopsy`; Fernandes et al., 2017), cirrhosis (`Status`, death vs. alive; Mayo Clinic trial data, Fleming & Harrington, 2013). Public, by OpenML identifier: `Support2` (46628), `rl` (41160), `Fraud_Detection_Dataset` (46357), `sick` (38), `colleges` (42121), `california_house_price` (46669), `SoilHydroDB` (42333), `SoilKsatDB` (42332), `Code_Smells_Data_Class` (43079), `Meta_Album_MD_MIX_Mini` (44287), `all-natural-disasters-19002021-eosdis` (46727), `fps_benchmark` (44992), `SAT11-HAND-runtime-regression` (41980), `BNP_Paribas_Cardif_Claims_Management` (46625), `ASP-POTASSCO-regression` (41704), `communities-and-crime-binary` (43891), `us_crime` (42730) and `albert_seed_0_nrows_2000_nclasses_10_ncols_100_stratify_True` (44538), each with its OpenML default target. Where a source appears on OpenML several times, we keep one copy: the screening pass that added the last five accepted ten datasets, of which five were near-duplicates of another accepted one (two further *Fraud-Detection* copies, three further *albert* subsamples and two further *communities-and-crime* variants). Numerical columns only; columns over 95% missing are dropped; rows capped at 5000. Cell-level missingness ranges from 5% to 37%.

Two decisions matter for validity. (i) Our three ADNI exports are successive releases of the same ADNIMERGE table rather than three cohorts, and the later release is a superset: every participant of the first two is present in the third (1736 and 1737 of its 2155), which carries 66 further columns. We therefore use the widest release only, taking one row per participant and the baseline (`_bl`) measurements. (ii) As distributed, ADNI and ABIDE contain the instruments that define their labels (ADNI: CDRSB, MMSE, ADAS, MOCA, FAQ, RAVLT, Ecog, mPACC and the current-visit diagnosis; ABIDE: ADOS, ADI-R, SRS, SCQ, AQ, Vineland). With them, mean imputation alone reaches AUC 0.999 and 0.998 and no imputation comparison is meaningful; we restrict ADNI to baseline biomarkers (MRI volumetrics, FDG and AV45 PET, CSF Aβ/tau/p-tau, APOE4, age, education, sex) and ABIDE to phenotype and image-quality measures, giving AUC 0.74–0.80 and 0.62–0.67. The 18 public datasets are downloaded by identifier; the four clinical cohorts are available from their custodians under their own access procedures, and our loaders document the target definitions, patient de-duplication and column exclusions applied here.

B PROTOCOL DETAILS

Tuning. Models and imputers not otherwise attributed are the scikit-learn implementations (Pedregosa et al., 2011). 50 Optuna trials per (dataset, seed, family) in the benchmark's search spaces, the benchmark's default MLP configuration enqueued first. Simulated data: tuned on the cell's complete data. Natural-missingness data: tuned on the mean-imputed training split, with the per-imputation control of §7.

Placement intervention and the degradation control. Gaussian error of pooled budget $B \in \{0.5, 1.0\}$ is added at the missing entries and allocated by one of five rules: uniform, proportional to permutation importance, inverse to importance, concentrated on the top third, or on the bottom third. The spans reported in §3 use all five. The degradation measure used as a control in §5 is the mean drop at $B = 1$ over four of them (uniform, proportional, top third and bottom third), matching the configuration battery, which does not include the inverse rule.

Definitions behind §3. Per-feature error e_j is the mean squared error of an imputer on the entries of feature j that are actually missing, computed on the training rows, so each feature's error is an average over its own missing entries. Two aggregations of $\{e_j\}$ appear. The *pooled* reconstruction error is the missing-entry-weighted mean $\sum_j m_j e_j / \sum_j m_j$ with m_j the number of missing entries in feature j, which is what an imputer is scored by; the intervention of §3 holds exactly this quantity fixed and varies only where the error sits. The *shares* reported in §3 and Table 9 instead aggregate the unweighted per-feature reductions, $\sum_{j \in \text{top}} \Delta_j / \sum_j \Delta_j$, so that a feature counts once regardless

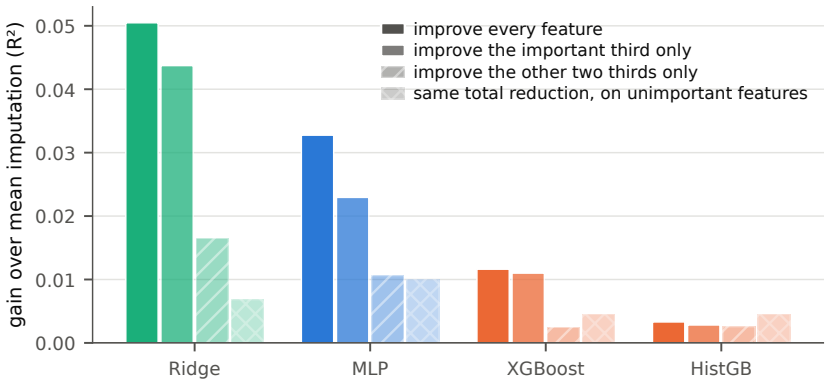


Figure 3: Attribution using the imputations themselves (MCAR 50%, medians over 19 datasets); arms and statistics as in Table 1, discussed in §3.

of how often it is missing; the importance shares $\sum_{j\in\text{top}} s_j$ use the same unweighted convention. The two coincide under MCAR, where m_j is constant in expectation, and can differ under MAR and MNAR, where missingness rates vary across features. The matched control of §3 is built on the unweighted reduction: features are added from the least important upwards until their summed Δ_j reaches that of the important third. Under MCAR this also matches the pooled reconstruction improvement the imputer is scored by; under MAR and MNAR it matches the placement budget but not necessarily that score, so the off-MCAR rows of Table 6 should be read as holding the per-feature reduction fixed rather than the imputer's own metric. The inverse allocation rule uses $1/\max(s_j, 10^{-6})$, so that features with zero estimated importance receive a large but finite share. The reduction is $\Delta_j = e_j^{\text{mean}} - e_j^{\text{best}}$, signed; in practice the better imputer is worse on no feature at the median, and recomputing every quantity with negative reductions clipped to zero changes the shares by at most 0.007. Importance s_j is the permutation importance of the model *as a practitioner would have it*: fitted on the mean-imputed training split, permuted on the validation split, five repeats, clipped at zero and normalised to sum to one. "Lift" is $\sum_j s_j\Delta_j$ divided by the value that allocation would take if Δ were spread uniformly, i.e. $\bar{\Delta}$. Appendix B states the variants we checked.

Imputer selection. On simulated data the best-reconstructing imputer is chosen by mean squared error on the *training rows'* actually missing entries; test rows are never used for selection. On real data it is chosen by error on an additional 10% of observed training entries sampled with a fixed seed, after which imputers are refitted on the data with its original missingness. Validation-based selection uses the 15% validation split, and the leave-one-dataset-out default of §8 is computed per target dataset from the remaining datasets only.

Capacity sweeps. MLP $(\text{depth}, \text{width}) \in \{(1,d),(2,2d),(3,4d),(4,8d),(6,\min(10d,1024))\}$; XGBoost `max_depth`$\in\{1,2,4,6,10\}$; random forest `max_depth`$\in\{2,6,\text{none}\}$.

C Model-Family Comparison

Table 4 pairs each neural family against each tree ensemble, and Figure 5 gives the per-family gains with bootstrap intervals, both for §5.

D Crossed Design: Absolute Scores

Figure 4 shows the four arms of §6 for every family and both mechanisms, and Table 5 gives the absolute scores behind the gains reported there.

Table 4: Tuned neural networks against tuned tree ensembles under simulated MCAR at 50%: median difference in gain, share of datasets favouring the neural side, Holm-corrected p (dataset-level Wilcoxon, $n = 19$ and $n = 15$), and the same difference on complete data (Δclean).

		Regression (R^2)			Classification (AUC)		
neural	tree	Δclean	Δgain	% / p_{Holm}	Δclean	Δgain	% / p_{Holm}
MLP	XGBoost	−0.006	+0.024	89% / .0008	−0.014	+0.012	93% / .0012
MLP	rand. forest	−0.005	+0.027	95% / .0002	−0.012	+0.015	87% / .0024
MLP	HistGB	−0.005	+0.030	95% / .0002	−0.010	+0.019	93% / .0012
FT-Transf.	XGBoost	+0.000	+0.018	79% / .0057	−0.001	+0.006	100% / .0007
FT-Transf.	rand. forest	+0.001	+0.017	79% / .0152	−0.000	+0.007	100% / .0007
FT-Transf.	HistGB	+0.003	+0.020	95% / .0003	−0.001	+0.010	87% / .0043
SAINT	XGBoost	+0.000	+0.021	84% / .0192	(untuned default only)		
SAINT	rand. forest	+0.001	+0.014	89% / .0035			
SAINT	HistGB	+0.004	+0.023	89% / .0029			

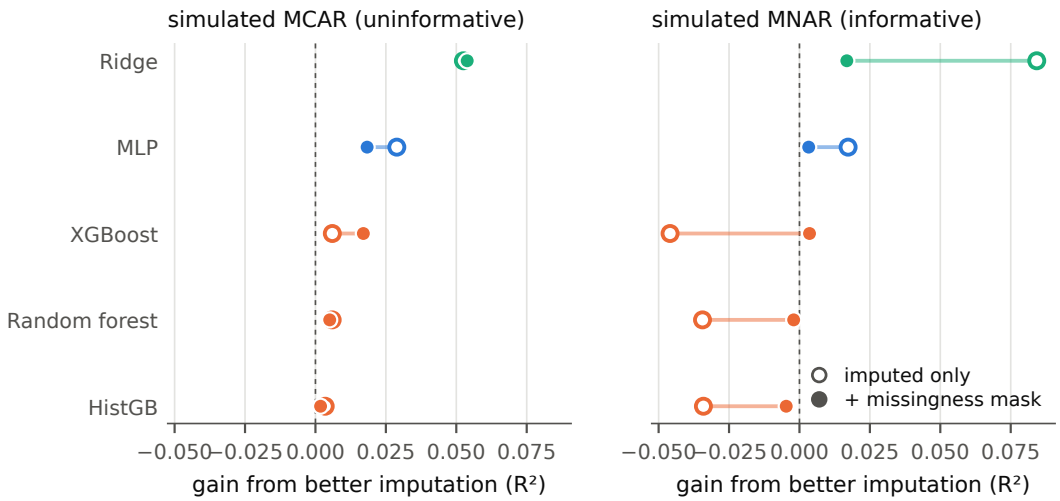


Figure 4: Crossed design, 19 datasets, tuned models, both mechanisms. Open circles: imputed features only; filled: the missingness mask supplied as extra features. Design following Le Morvan & Varoquaux (2025), Appendix L.2; §6 states what we add.

E Additional Results

Equivalence, not just a null test. For each (model, imputer) pair of Table 2 we bootstrap the mean paired difference between the targeted and full pipelines over the 19 datasets and ask whether the 90% interval lies inside ± 0.01 R^2. It does for the random-forest imputer with HistGB ($+0.0009$, $[-0.0043, +0.0057]$) and with XGBoost ($+0.0005$, $[-0.0056, +0.0063]$). In three pairs the interval excludes zero in the targeted pipeline's favour (iterative imputer with HistGB $+0.0116$ and XGBoost $+0.0107$) or against it (random-forest imputer with the MLP -0.0126), and in the remaining three it is too wide to decide. We therefore report no detected loss rather than equivalence.

Restricting the columns or dropping the iteration. The arms behind §4's decomposition, on the 19 regression datasets at MCAR 50% with the iterative imputer: `full` runs ten rounds over every column, `targ_pass` one pass over the top third, `targ_iter` the same ten rounds but updating only the top third, and `rank_random` the same as `targ_iter` with a random subset of equal size. Median gains over mean imputation are, for HistGB, -0.0115, $+0.0004$, $+0.0005$ and -0.0072; for XGBoost, -0.0046, $+0.0025$, -0.0050, -0.0043; and for ridge, $+0.0248$, $+0.0329$, $+0.0168$, $+0.0041$, in that order.

Besides the tables cited from the main text, Table 7 varies the targeting budget on the regression suite and Table 13 tests the importance-weighted reconstruction metric as a predictor of the gain.

Table 5: Absolute test R^2 (median over 19 datasets, 5 seeds) for the four arms of the crossed design. Gains are medians of per-dataset paired differences and need not equal the differences between the medians shown in this table.

mech.	model	mean	best	mean+mask	best+mask	gain w/o mask	gain w/ mask
MCAR	ridge	0.303	0.415	0.302	0.414	+0.053	+0.054
MCAR	MLP	0.430	0.508	0.458	0.512	+0.029	+0.018
MCAR	XGBoost	0.485	0.474	0.486	0.480	+0.006	+0.017
MCAR	rand. forest	0.472	0.471	0.481	0.471	+0.006	+0.005
MCAR	HistGB	0.484	0.474	0.491	0.475	+0.004	+0.002
MNAR	ridge	0.295	0.386	0.478	0.494	+0.084	+0.017
MNAR	MLP	0.448	0.516	0.573	0.572	+0.017	+0.003
MNAR	XGBoost	0.550	0.532	0.554	0.562	−0.046	+0.004
MNAR	rand. forest	0.572	0.537	0.571	0.567	−0.034	−0.002
MNAR	HistGB	0.563	0.542	0.569	0.564	−0.034	−0.005

Table 6: The attribution experiment outside its original regime: medians over datasets, and the paired contrast between the important-third arm and the matched-reduction control (dataset-level Wilcoxon, Holm-corrected within each block). The MCAR-50% regression block is Table 1.

regime	model	gain (all)	gain (top third)	gain (matched ctrl.)	top − ctrl.	p_{Holm}
MCAR 20%, reg. (19 datasets)	Ridge	+0.0530	+0.0478	+0.0054	+0.0419	$< 10^{-4}$
	MLP	+0.0499	+0.0278	+0.0060	+0.0179	0.0006
	XGBoost	+0.0157	+0.0162	+0.0050	+0.0077	0.0048
	HistGB	+0.0187	+0.0128	+0.0016	+0.0125	0.0048
MCAR 50%, clf. (15 datasets, AUC)	Logistic	+0.0146	+0.0093	+0.0024	+0.0067	0.0498
	MLP	+0.0144	+0.0107	+0.0035	+0.0035	0.563
	XGBoost	−0.0032	−0.0025	+0.0018	−0.0017	0.563
	HistGB	−0.0025	−0.0009	−0.0004	−0.0030	0.600
MNAR 50%, reg. (19 datasets)	Ridge	+0.0842	+0.0515	+0.0222	+0.0067	0.120
	MLP	+0.0173	+0.0144	+0.0250	−0.0083	0.120
	XGBoost	−0.0459	−0.0223	+0.0030	−0.0219	0.0114
	HistGB	−0.0341	−0.0256	−0.0046	−0.0201	0.0247

Table 7: (A) Targeted imputation across budgets, MCAR 50% regression, medians over 19 datasets. "net" includes the importance ranking. The gain keeps rising slowly with the budget while the saving falls, so $k = d/3$ sits at the knee.

	gain (random-forest imputer)			net speed-up		
budget k/d	Ridge	XGBoost	HistGB	Ridge	XGBoost	HistGB
1/6	+0.038	+0.002	−0.000	23.6×	10.2×	6.4×
1/3	+0.065	+0.006	+0.002	12.3×	5.3×	5.8×
1/2	+0.069	+0.005	−0.001	8.2×	6.1×	2.6×
2/3	+0.072	+0.010	+0.001	6.9×	5.0×	4.3×

Robustness of the decomposition and the attribution. The quantities in §3 depend on how importance is measured and on how negative reductions are treated; both were checked. Recomputing every share with negative reductions clipped to zero changes them by at most 0.007 (the better imputer is worse on no feature at the median, so there is almost nothing to clip). Repeating the attribution experiment with importance taken from a model fitted to the *complete* data instead of the mean-imputed data leaves the conclusion intact: the important-third arm still delivers 82–85% of the full gain for ridge and HistGB (against 85–87%), and the contrast against the matched-reduction arm is +0.0143 R^2 for ridge ($p < 10^{-4}$) against +0.0153 with the practitioner's importance. The two references differ mainly in how concentrated importance looks (84–94% in the top third on complete data, 72–84% after mean imputation), not in what the attribution shows.

Table 8: (C) Targeted imputation on the classification suite and on natural missingness: median gain over mean imputation for the full and targeted pipelines, their paired difference, and the net speed-up including the ranking step. All natural-missingness analyses in the paper use the same 23 datasets.

setting	model	imputer	full	targeted	diff. (targ.−full), (p)	net
classification	HistGB	random forest	-0.0014	$+0.0002$	$+0.0052$ (0.018)	$8.9\times$
(15 datasets,	XGBoost	random forest	-0.0073	$+0.0006$	$+0.0066$ (0.003)	$9.7\times$
AUC)	Logistic	random forest	$+0.0150$	$+0.0126$	-0.0005 (0.68)	$11.3\times$
	Random forest	iterative	-0.0121	-0.0024	$+0.0069$ (0.008)	$0.1\times$
natural	Random forest	random forest	-0.0035	-0.0005	$+0.0016$ (0.003)	$3.0\times$
missingness	HistGB	random forest	-0.0021	-0.0002	$+0.0014$ (0.142)	$7.4\times$
(23 datasets)	XGBoost	random forest	-0.0022	-0.0016	$+0.0012$ (0.056)	$7.5\times$
	Linear	random forest	$+0.0014$	$+0.0022$	-0.0010 (0.345)	$8.8\times$

Table 9: Where a real imputer's error reduction lands (medians over 19 datasets, 5 seeds). "reduction in top third": the share of the total per-feature error reduction (best-reconstructing imputer over mean imputation) falling on the third of features with the highest permutation importance; "importance in top third": the share of importance those features carry; "lift": the importance-weighted reduction divided by what an importance-blind allocation would give. Importance here comes from a model fitted to the *complete* data, as in Figure 1, not from the mean-imputed model of Table 1, which is why the shares differ.

model	reduction in top third	importance in top third	lift
Ridge	0.326	0.862	1.13
MLP	0.345	0.843	1.06
SAINT	0.327	0.856	1.00
XGBoost	0.330	0.945	1.02
HistGB	0.326	0.940	1.03

Table 10: Natural-missingness datasets, all 23: median gain over mean imputation under four ways of choosing the imputer. All four rules draw on the same four candidates (mean, KNN, iterative, random-forest); "valid." restricts the choice to the three non-mean candidates while "valid.+mean" and "oracle" allow mean to win, which makes the oracle non-negative by construction. Twelve regression and eleven classification datasets, 5 seeds.

	Regression (12 datasets, R^2)				Classification (11 datasets, AUC)			
model	recon.	valid.	valid.+mean	oracle	recon.	valid.	valid.+mean	oracle
linear	$+.001$	$+.001$	$+.001$	$+.007$	$+.001$	$+.005$	$+.004$	$+.011$
MLP	$-.006$	$-.002$	$+.000$	$+.005$	$+.001$	$+.002$	$+.002$	$+.009$
FT-Transformer	$-.009$	$-.005$	$-.000$	$+.004$	$-.001$	$-.000$	$+.001$	$+.010$
XGBoost	$-.000$	$-.000$	$+.000$	$+.005$	$-.003$	$+.000$	$+.000$	$+.005$
random forest	$-.001$	$-.002$	$+.000$	$+.003$	$-.004$	$-.008$	$-.000$	$+.004$
HistGB	$-.001$	$-.000$	$+.001$	$+.006$	$-.002$	$-.006$	$+.000$	$+.007$

Effect sizes for the rejected interventions of §9. Stochastic draws from the imputation posterior lose a median 0.021 R^2 against the corresponding conditional-mean imputation and variance correction 0.014, each worse on 75–100% of cells ($p < 10^{-5}$, 19 datasets, four model families, three mechanisms). Sensitivity-weighted allocation of an imputation budget beats a random allocation by $+0.0085$ R^2 for ridge but is indistinguishable for gradient-boosted trees ($+0.002$, $p = 0.42$). A sensitivity$\times$imputability product loses to sensitivity alone in all four model families. Mask-interaction features are worth $+0.0035$ on average, with the effect confined to two datasets where iterative imputation diverges. Target-aware imputation degrades reconstruction (pooled MSE $0.73 \to 0.97$) and prediction at once.

Placement spans by rate and mechanism. Median span of downstream R^2 across placements at identical pooled reconstruction error: 0.061–0.090 at 20% MCAR, 0.107–0.173 at 50% MCAR,

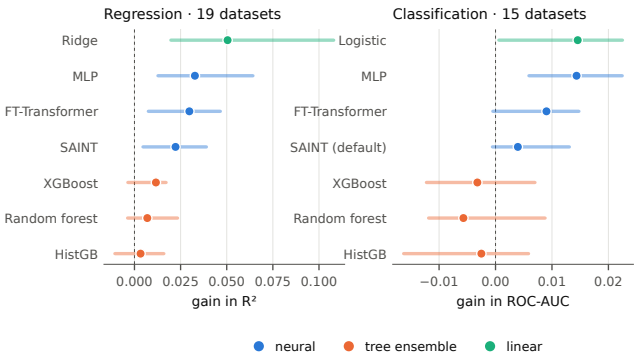


Figure 5: Gain from better imputation under simulated MCAR at 50%, medians over datasets with 95% bootstrap intervals (§5).

Table 11: The attribution under five feature rankings (19 regression datasets, MCAR 50%, medians; paired dataset-level Wilcoxon). "overlap" is the share of the top third that the ranking has in common with the permutation ranking used elsewhere.

model	ranking	top third	matched ctrl.	top − ctrl.	p	overlap
Ridge	permutation	+0.0437	+0.0068	+0.0153	$< 10^{-4}$	—
	drop-column	+0.0385	+0.0082	+0.0151	$< 10^{-4}$	0.80
	conditional permutation	+0.0433	+0.0045	+0.0153	0.0001	0.93
	native (gain / $\lvert\beta\rvert$)	+0.0430	+0.0042	+0.0141	$< 10^{-4}$	0.90
	random	+0.0180	+0.0201	−0.0036	0.11	0.28
XGBoost	permutation	+0.0110	+0.0045	+0.0036	0.26	—
	drop-column	+0.0135	+0.0086	+0.0013	0.95	0.66
	conditional permutation	+0.0095	+0.0078	+0.0032	0.37	0.91
	native	+0.0130	+0.0062	+0.0054	0.17	0.80
	random	+0.0039	+0.0066	−0.0021	0.92	0.30
HistGB	permutation	+0.0028	+0.0045	+0.0006	0.77	—
	drop-column	+0.0028	+0.0052	−0.0021	0.44	0.70
	conditional permutation	+0.0028	+0.0035	+0.0007	0.83	0.90
	native	+0.0006	+0.0035	+0.0001	0.89	0.37
	random	+0.0021	+0.0030	+0.0010	0.33	0.30

Table 12: Tree ensembles on the 23 natural-missingness datasets: median change relative to mean imputation, and the paired contrast between native handling and targeted imputation (§8).

task	model	selected imputer	targeted	mean+indicator	native	native − targeted (p)
clf (11)	HistGB	−0.0021	−0.0035	+0.0012	+0.0015	+0.0041 (0.024)
	XGBoost	−0.0029	−0.0016	+0.0001	+0.0015	+0.0030 (0.032)
reg (12)	HistGB	−0.0010	+0.0004	+0.0024	+0.0033	+0.0014 (0.34)
	XGBoost	−0.0002	−0.0013	−0.0004	+0.0027	+0.0053 (0.052)

0.035–0.052 at 50% MAR, 0.106–0.148 at 50% MNAR (ranges over ridge, MLP, SAINT, HistGB, XGBoost), exceeding 0.10 in 21–72% of cells.

Missingness rate. Rerunning the simulated regression study at 20% leaves the family effect intact and slightly larger at matched accuracy (+0.0257, [+0.0117, +0.0409]); gains grow for every family, since conditional-mean imputation is more accurate when more of each row is observed.

Rejected explanations for the tree exception under MCAR. Trees do saturate (prediction displacement per unit input perturbation falls from 0.91 to 0.36 for XGBoost) but so does SAINT (0.54 → 0.32), which benefits from better imputation; normalised gains remain about three times

Table 13: Does weighting the reconstruction metric by importance explain gains better than the pooled metric? Spearman correlation across 19 datasets between each quantity and the observed gain, with a 4000-resample bootstrap interval for the difference. No difference is resolvable.

model	pooled ρ	importance-weighted ρ	difference [95% CI]
Ridge	$+0.591$	$+0.628$	$+0.037\ [-0.175, +0.246]$
MLP	$+0.472$	$+0.502$	$+0.030\ [-0.207, +0.232]$
SAINT	$+0.279$	$+0.275$	$-0.004\ [-0.335, +0.322]$
XGBoost	$+0.454$	$+0.474$	$+0.019\ [-0.379, +0.369]$
HistGB	$+0.421$	$+0.558$	$+0.137\ [-0.225, +0.519]$

lower for trees after accounting for headroom; dataset redundancy does not correlate with the gain ($p \in [0.24, 0.75]$); the rank correlation between sensitivity and true single-feature damage is slightly lower for every non-linear model, not specifically trees (only the MLP contrast survives Holm); and damage is super-additive for every family, more so for trees (1.72 vs. 1.34 for ridge, $p = 0.011$), the opposite of the sub-additivity hypothesis.

F Upstream Defects

SAINT on GPU. The reference implementation allocates an intermediate tensor without a device argument; we pass the input's device and dtype. **MAR mask generation.** Logistic MAR masks drawn on the raw feature scale drive six of the 19 regression datasets to a fully missing column; we draw them on training-standardised features.